

Using Invariant Detection Mechanism in Black Box Inference

Muzammil Shahbaz* and Roland Groz**

*France Telecom R&D
Meylan, France

Muhammad.MuzammilShahbaz@orange-ftgroup.com

**LIG, Computer Science Lab
Grenoble Universités, France
Roland.Groz@imag.fr

Abstract. The testing and formal verification of black box software components is a challenging domain. The problem is even harder when specifications of these components are not available. An approach to cope with this problem is to combine testing with learning techniques, such that the learned models of the components can be used to explore unknown implementation and thus facilitate testing efforts. In recent years, we have contributed to this approach by proposing techniques for learning parameterized state machine models and then use them in the integration testing of black box components. The major problem in this technique left unaddressed was the selection of parameter values during the learning process. In this paper, we propose to use an invariant detection mechanism to select values in the learning process, thus refining model inference and testing approach. Initial experiments with small examples yielded positive results.

1 Introduction

Although formal methods are a key element to automate a number of phases of software development, very few software processes are actually fully based on formal methods and associated tools. Typical problems arise when specifications are not formal or only a part of the specifications is formalized. It is also often the case that formal models are not updated consistently with software evolution.

One approach to alleviate this problem is to revert the usual rigid process that starts from formal specifications. This flexible approach consists in introducing or re-introducing formal models during the software process, typically by retrieving them from various sources, including the code itself. The advantage is that formal methods and tools can be applied on development steps where they can be justified as more efficient and therefore more acceptable for integration in the development process of the company. This approach is commended especially in a testing phase which is a cost-intensive and a time consuming activity and thus automation is highly desirable. Formal methods can support testing through test generation, test interpretation, classification and diagnosis.